

WHAT IS CLAIMED IS:

1. A method of binding processes in a network system, the method comprising:
 - binding a parent process with a remote method invocation process; and
 - calling an object associated with the parent process, the object initiating a thread to perform the steps of:
 - monitoring the status of remote method invocation processes; and
 - rebinding the parent process with an active remote method invocation process when the thread determines that its parent process is not bound with an active remote method invocation process.
2. The method of claim 1, wherein the binding step comprises:
 - providing a network address of the parent process to the active remote method invocation process.
3. The method of claim 1, wherein the monitoring step comprises:
 - performing a list call to an active remote method invocation process to determine whether the parent process is bound to an active remote method invocation process.
4. The method of claim 3, wherein the rebinding step comprises:
 - performing a rebind call to an active remote method invocation process.
5. The method of claim 1, wherein the monitoring step comprises:

1 calling an active remote method invocation process to determine whether the parent
2 process network address is registered with an active remote method invocation process.

3
4 6. The method of claim 1, comprising:

5 binding a second parent process with a remote method invocation remote method
6 invocation process; and

7 calling a second object associated with the second parent process, the second object
8 initiating a second thread to perform the steps of:

9 monitoring the status of remote method invocation processes; and

10 rebinding the second parent process with an active remote method
11 invocation process when the second thread determines that the second parent
12 process is not bound with an active remote method invocation process.

13
14 7. The method of claim 1, wherein the step of binding a parent process comprises:

15 binding one of an RMI daemon, a distributed task facility daemon, a log manager
16 daemon, or a domain manager daemon, with an active RMI daemon.

17
18 8. The method of claim 1, comprising:

19 terminating the thread when the parent process is terminated.

20
21 9. A network system, comprising:

22 a plurality of remote nodes, at least one of the remote nodes running a remote method
23 invocation process; and

1 a management server for managing the remote nodes, the management server
2 including at least one processor for running an remote method invocation process and at least
3 one management process, each at least one management process being associated with an
4 object capable of initiating a thread to perform the steps of:

5 monitoring the status of remote method invocation processes; and
6 rebinding the at least one management process with an active remote method
7 invocation process when the thread determines that the at least one management
8 process is not bound with an active remote method invocation process.

9
10 10. The network system of claim 9, wherein the at least one management process
11 comprises a plurality of management processes.

12
13 11. The network system of claim 9, wherein the plurality of management
14 processes comprise:

15 a distributed task facility process;
16 a domain manager process; and
17 a log manager process.

18
19 12. The network system of claim 9, wherein each of the remote nodes runs a
20 service control manager agent process for performing server management tasks.

21
22 13. The network system of claim 9, wherein the management server comprises:
23 a secondary storage device, the secondary storage device comprising:
24 a data repository;

1 a depot; and

2 a web server.

3
4 14. The network system of claim 9, wherein the plurality of remote nodes are
5 arranged into at least one node group, the network system comprising a service control
6 manager for managing the at least one node group.

7
8 15. A method of binding a parent process to an remote method invocation process,
9 the method comprising:

10 a) performing a rebind call to an remote method invocation process to provide a
11 network address and an object interface of a parent process to the remote method invocation
12 process; and

13 b) performing an initialization call to an object associated with the parent process, the
14 initialization call initiating a thread, the thread performing the steps of:

15 1) performing a list call to an active remote method invocation process to
16 determine whether the parent process is bound with the active remote
17 method invocation process;

18 2) performing a rebind call to an active remote method invocation process if
19 the thread determines that the parent process is not bound with an active
20 remote method invocation process; and

21 3) repeating steps 1 and 2.

1 16. The method of claim 15, wherein the parent process is one of a remote method
2 invocation daemon, a distributed task facility daemon, a log manager daemon, and a domain
3 manager daemon.

4
5 17. The method of claim 15, wherein the step of performing a rebind call includes the
6 step of performing a rebind call to a remote method invocation daemon.

7
8 18. The method of claim 15, wherein the step of performing a list call includes the
9 step of performing a list call to a remote method invocation daemon.

10
11 19. The method of claim 15, comprising:

12 terminating the thread when the parent process is terminated.